Modeling of Hurricane Impacts

Interim Report 2
June – August 2006

Dano Roelvink (P.I.), Ad Reniers,
Ap van Dongeren, Dirk-Jan Walstra

UNESCO-IHE Institute for Water Education (Contractor)
WL | Delft Hydraulics
Delft University of Technology

Contract no. N62558-06-C-2006

The Research reported in this document has been made possible through the support and sponsorship of the U.S. Government through its European Research Office of the U.S. Army.
Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*
01-09-2006

2. REPORT DATE
Interim report

3. DATES COVERED *(From - To)*
June-August 2006

4. TITLE AND SUBTITLE
Modeling of
Hurricane Impacts,
Interim Report 2

5a. CONTRACT NUMBER
N62558-06-C-2006

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Roelvink, Dano
Reniers, Ad
van Dongeren, Ap
Walstra, Dirk-Jan

5d. PROJECT NUMBER

5e. TASK NUMBER
0002

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
UNESCO-IHE INSTITUTE FOR WATER EDUCATION
PO BOX 3015
2601 DA DELFT
NETHERLANDS

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
FISC SIGONELLA
NAVAL REGIONAL CONTRACTING DET LONDON
SHORE/FLEET TEAM
BLOCK 2, WING 12, GOVT BLDGS LIME GROVE
RUISLIP MIDDLESEX HA4 8BX
UNITED KINGDOM

10. SPONSOR/MONITOR'S ACRONYM(S)
ERDC

11. SPONSORING/MONITORING AGENCY REPORT NUMBER

12. DISTRIBUTION AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES
Prepared in collaboration with WL | Delft Hydraulics and Delft University of Technology

14. ABSTRACT
This first interim report describes the development of dune erosion algorithms, based on an analysis of existing approaches; it describes the activities related to the ShoreCirc model required to make it suitable for modeling of the nearshore hydrodynamics during hurricanes, and it describes a newly developed model for inner surfzone, swash and overwash processes. Significant progress was made here and a set of Matlab routines providing much of the needed functionality is included for evaluation and testing. A novel approach for solving the time-varying wave action balance is applied, where, in contrast with existing surfbeat models which are averaged over directional and frequency spectrum, the wave directionality is maintained, which removes the need for a separate wave model to provide wave directions.

15. SUBJECT TERMS
Surf zone, swash, overwash, wave groups, wave propagation, morphology, dune erosion

16. SECURITY CLASSIFICATION OF: a. REPORT	b. ABSTRACT	c. THIS PAGE	17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
unclassified	unclassified	unclassified	unclassified	15	Prof. Dano Roelvink

19b. TELEPONE NUMBER (*Include area code)*
+31 15 2151838

Contents

Abstract

Over the last months, much effort has been put into further development of the XBeach model. A visit was made to Delaware by Ap van Dongeren, to consult with Brad Johnson of ERDC about the coupling between XBeach and Shorecirc; it was decided to focus on further (joint) development of XBeach. The following improvements and extensions were carried out:

- Improvement of numerical scheme in line with Stelling and Duinmeijer method, to improve long-wave runup and backwash on the beach. The momentum-conserving form is applied, while retaining the simple first-order approach. The resulting scheme has been tested against the well-known Carrier and Greenspan test.
- Implementation of the Generalised Lagrangean Mean (GLM) approach to represent the depth-averaged undertow and its effect on bed shear stresses and sediment transport, cf. Reniers et al. (2004)
- Implementation of Roelvink (1993) wave dissipation model for use in the instationary wave energy balance (in other words, when the wave energy varies on the wave group timescale)
- Implementation of Soulsby – Van Rijn transport formulations, cf Reniers et al (2004).
- Automatic time step based on Courant criterion, with output at fixed time intervals.
- Improvement of avalanching mechanism, with separate criteria for critical slope at wet or dry points.

A first series of (mainly 1D) validation tests have been carried out, culminating in the successful simulation of a dune erosion test carried out in a large-scale wave flume (LIP11D test 2E, Arcilla et al., 1993). The results are briefly shown in this report.

1 Introduction

This report is the second progress report of the project ‘Modeling of Hurricane Impacts’, contract no. N62558-06-C-2006, which was granted by the US Army Corps of Engineers, Engineer Research and Development Center (ERDC), European Research Office and administered by FISC SIGONELLA, NAVAL REGIONAL CONTRACTING DET LONDON, SHORE/FLEET TEAM.

The project is being carried out by Prof. Dano Roelvink of UNESCO-IHE (Principal Investigator), Dr. Ad Reniers and Jaap van Thiel de Vries of Delft University of Technology and Dr. Ap van Dongeren, Dirk-Jan Walstra and Jamie Lescinski of WL | Delft Hydraulics.

Over the last months, much effort has been put into further development of the XBeach model. A visit was made to Delaware by Ap van Dongeren, to consult with Brad Johnson of ERDC about the coupling between XBeach and Shorecirc; it was decided to focus on further (joint) development of XBeach.

A number of improvements and extensions were made; they are described in Chapter 3. Chapter 4 contains an update of all formulations used and their numerical schematization. In

Chapter 5, a series of validation tests is described, from analytical experiments to check the numerical behaviour to a large-scale dune erosion experiment in a wave flume (Arcilla et al, 1993). In Chapter 6 we draw conclusions and sketch our plans for the coming period.

2 Visit Ap van Dongeren to Delaware

In a two-day visit by Ap van Dongeren to the University of Delaware, he discussed progress of XBeach and ShoreCirc with Brad Johnson. Given the overlapping functionality in the surf zone and the dedicated functionality of XBeach in the swash zone it was decided to not interface with Shorecirc, but rather to incorporate elements of Brad's code into XBeach. To this end, the source code of XBeach that was delivered after the first three months was discussed in detail, including more recent updates.

3 XBeach model development and validation

3.1 Status in June 2006

The main objective of the XBeach model is to provide a robust and flexible environment in which to test morphological modelling concepts for the case of dune erosion, overwashing and breaching. The top priority is to provide numerical stability; first order accuracy is accepted since there is a need for small space steps and time steps anyway, to represent the strong gradients in space and time in the nearshore and swash zone. Because of the many shock-like features in both hydrodynamics and morphodynamics we choose upwind schematizations as a means to avoid numerical oscillations which can be deadly in shallow areas.

The modelling environment should be flexible and the code easy to comprehend and concise; therefore we have adapted the Matlab environment as development environment; conversion to Fortran 90/95 at a later stage will be straightforward.

The code has the following functionality:

- Depth-averaged shallow water equations including time-varying wave forcing terms; combination of sub- and supercritical flows;
- Time-varying wave action balance including refraction, shoaling, current refraction and wave breaking;
- Wave amplitude effects on wave celerity;
- Depth-averaged advection-diffusion equation to solve suspended transport;
- Bed updating algorithm including possibility of avalanching;
- Possibility to extend to parallel multi-domain version;

3.2 Improvements and extensions June-August 2006

The following improvements and additions have been implemented and tested in the last three months:

- Improvement of numerical scheme in line with Stelling and Duinmeijer method, to improve long-wave runup and backwash on the beach. The momentum-conserving form is applied, while retaining the simple first-order approach. The resulting scheme has been tested against the well-known Carrier and Greenspan test.
- Implementation of the Generalised Lagrangean Mean (GLM) approach to represent the depth-averaged undertow and its effect on bed shear stresses and sediment transport, cf. Reniers et al. (2004)
- Implementation of Roelvink (1993) wave dissipation model for use in the instationary wave energy balance (in other words, when the wave energy varies on the wave group timescale)
- Implementation of Soulsby – Van Rijn transport formulations, cf Reniers et al (2004).
- Automatic time step based on Courant criterion, with output at fixed time intervals.
- Improvement of avalanching mechanism, with separate criteria for critical slope at wet or dry points.

3.3 General setup (update)

The program XBeach consists of a main Matlab *script*, xbeach.m, and a number of *functions* that operate on two *structures*:

- par – this contains general input parameters
- s - this contains all the arrays for a given computational domain

For a single-domain run, one structure s is passed between flow, wave, sediment and bed update solvers, which extract the arrays they need from the structure elements to local variables, do their thing and pass the results back to the relevant structure elements. This makes the overall program clear, prevents long parameter lists and makes it easy to add input variables or arrays where needed.

For multi-domain runs, one can define multiple instantiations of the structure s which are passed to the same functions; an additional function is needed to pass the boundary information between the domains back and forth. We have carried out a simple test of this principle, without actually implementing a multi-processor version, which confirms that the data structure can handle this case.

In the Table 1 we will outline the various functions and their purposes. The main script xbeach.m is reproduced in Table 2 below.

Function call	Purpose
[par] = wave_input	Creates elements of structure par containing wave input

	parameters
[par] = flow_input(par)	Adds elements of structure par containing flow input parameters
[par] = sed_input(par)	Adds elements of structure par containing sediment input parameters
[s] = grid_bathy;	Creates grid and bathymetry and stores them in structure s
[s] = wave_dist (s,par);	Creates initial directional spectrum at sea boundary
[s,par]=wave_init (s,par);	Initialises arrays (elements of s) for wave computations
[s] = flow_init (s,par);	Initialises arrays (elements of s) for flow computations
[s] = sed_init (s);	Initialises arrays (elements of s) for sediment computations
[s] = wave_bc (s,par,it);	Wave boundary conditions update, each timestep
[s] = flow_bc (s,par,it);	Flow boundary conditions update, each timestep
[s]=wave_timestep (s,par);	Carries out one wave timestep
[s]=flow_timestep (s,par);	Carries out one flow timestep
[s]=transus(s,par);	Carries out one suspended transport timestep
[s]=bed_update(s,par);	Carries out one bed level update timestep
output(it,s,par)	Performs output

Table 1 Overview of Matlab functions XBeach

Of these functions, the input functions and grid_bathy are case-dependent at present and actually contain the input values and grid and bathymetry definitions. This can however easily be replaced by functions with the same output, which read data from input files or input them through a screen dialog. We aim to leave the initialisation, boundary conditions and computational functions case-independent. The output function can be adapted to fit specific needs.

```
clear all
%
% General input per module
[par] = wave_input;
[par] = flow_input(par);
[par] = sed_input(par);
% Grid and bathymetry
[s] = grid_bathy;
% Directional distribution wave energy
[s] = wave_dist (s,par);
% Initialisations
[s,par] = wave_init (s,par);
[s]     = flow_init (s,par);
[s,par] = sed_init (s,par);
nt=par.nt;
par.tnext=par.tint;
it=0;
while par.t<par.tstop;
    % Wave boundary conditions
    [par,s] = wave_bc (s,par,it);
    % Flow boundary conditions
    [s] = flow_bc (s,par,it);
    % Wave timestep
    [s] = wave_timestep (s,par);
    % Flow timestep
    [s,par] = flow_timestep (s,par);
    % Suspended transport
    [s]=transus(s,par);
    % Bed level update
    [s]=bed_update(s,par);
    % Output
    [it,s]=output(it,s,par);
End
```

Table 2 Main routine xbeach.m

3.4 Overview of modifications Matlab code

Routine	**Modifications June-August 2006 (from v000 to v010)**
Baldock	None
Bed_update	Avalanching Morphological factor
Dispersion	None
Flow_bc	Timestep management
Flow_int	Include initial water level Initialize qx,qy
Flow_input	Added initial water level, Tstart, Tint, Tstop
Flow_timestep	GLM velocities Stelling and Duinmeijer scheme (momentum-conserving, first order) Bug-fix seaward boundary Automatic timestep
Grid_bathy	Model dependent
Output	Model dependent
Roelvink	New, computes dissipation according to Roelvink (1993)

Sb_vr	New; port formulations according to Soulsby- van Rijn
Sed_init	None
Sed_input	4 parameters added
Slope	None
Transus	Eulerian velocities used Equil. Concentration according to Soulsby- van Rijn
Wave_bc	Model-dependent treatment of wave boundary conditions (e.g. energy time-series read from file)
Wave_init	Intriduced initial water level Set initial value time step
Wave input	Added parameters Roelvink ’93 formulation Added choice of formulation
Wave_timestep	Added Roelvink ’93 formulation Computation of Urms, Ustokes
Xbeach	Time management introduced

3.5 Conversion to Fortran 90/95

The reason to work on a Fortran version of the code is that it is easier to create stand-alone versions on any platform, the code may be easier to parallellize and it may be faster.

Jamie Lescinski at Delft Hydraulics has converted the Matlab code to Fortran90. She is trying two tracks:

- to convert the Fortran code to MEX files and call them from a Matlab environment, This is posing some problems that may be due to a bug in Matlab, which she is presently trying to solve with the Matlab help desk;
- to convert all code to Fortran. This has just been completed and is ready to be tested.

The conversion from Matlab to Fortran appears to be relatively easy, as the ‘derived type’ in Fortran 90/95 is very similar to Matlab structures.

4 XBeach formulations (update)

4.1 Wave action equation solver

The wave forcing in the shallow water momentum equation is obtained from a time dependent version of the wave action balance equation. Similar to Delft University's HISWA model, the directional distribution of the action density is taken into account whereas the frequency spectrum is represented by a single mean frequency. The wave action balance is then given by:

$$\frac{\partial A}{\partial t}+\frac{\partial c_x A}{\partial x}+\frac{\partial c_y A}{\partial y}+\frac{\partial c_\theta A}{\partial \theta}=-\frac{D}{\sigma} \tag{4.1}$$

with the wave action:

$$A(x,y,\theta)=\frac{S_w(x,y,\theta)}{\sigma(x,y)} \tag{4.2}$$

where S_w represents the wave energy in each directional bin and σ the intrinsic wave frequency. The wave action propagation speeds in x- and y-direction are given by:

$$\begin{aligned} c_x(x,y,\theta)&=c_g(x,y)\bullet\cos(\theta)+u(x,y)\\ c_y(x,y,\theta)&=c_g(x,y)\bullet\sin(\theta)+v(x,y)\end{aligned} \tag{4.3}$$

where θ represents the angle of incidence with respect to the x-axis. The propagation speed in θ-space is obtained from:

$$\begin{aligned} c_\theta(x,y,\theta)=\frac{\sigma}{\sinh 2kh}\left(\frac{\partial h}{\partial x}\sin\theta-\frac{\partial h}{\partial y}\cos\theta\right)+\cos\theta\left(\sin\theta\frac{\partial u}{\partial x}-\cos\theta\frac{\partial u}{\partial y}\right)+\\ +\sin\theta\left(\sin\theta\frac{\partial v}{\partial x}-\cos\theta\frac{\partial v}{\partial y}\right)\end{aligned} \tag{4.4}$$

taking into account bottom refraction (first term on the RHS) and current refraction (last two terms on the RHS). The wave number k is obtained from the eikonal equations:

$$\begin{aligned} \frac{\partial k_x}{\partial t}+\frac{\partial \omega}{\partial x}=0\\ \frac{\partial k_y}{\partial t}+\frac{\partial \omega}{\partial y}=0\end{aligned} \tag{4.5}$$

where the subscripts refer to the direction of the wave vector components and ω represents the absolute radial frequency. The wave number is the obtained from:

$$k = \sqrt{k_x^2 + k_y^2} \tag{4.6}$$

The absolute radial frequency is given by:

$$\omega = \sigma + \vec{k} \bullet \vec{u} \tag{4.7}$$

and the intrinsic frequency is obtained from the linear dispersion relation:

$$\sigma = \sqrt{gk \tanh kh} \tag{4.8}$$

The group velocity is obtained from linear wave theory:

$$c_g = nc = \left(\frac{1}{2} + \frac{kh}{\sinh 2kh} \right) \frac{\sigma}{k} \tag{4.9}$$

This concludes the advection of wave action. The wave energy dissipation due to wave breaking is modelled according to Baldock et al. [1998]:

$$\bar{D} = \frac{1}{4} \alpha Q_b \rho g f_m \left(H_b^2 + H_{rms}^2 \right) \tag{4.10}$$

with $\alpha = O(1)$ and f_m representing the mean intrinsic frequency. The fraction of breaking waves is given by:

$$Q_b = \exp\left[-\left(\frac{H_b^2}{H_{rms}^2} \right) \right] \tag{4.11}$$

where the breaking wave height is:

$$H_b = \frac{0.88}{k} \tanh\left[\frac{\gamma kh}{0.88} \right] \tag{4.12}$$

and γ is a calibration parameter. The root mean square wave height is obtained from:

$$H_{rms} = \sqrt{\frac{8\int S_w(x,y,\theta) d\theta}{\rho g}} = \sqrt{\frac{8E_w}{\rho g}} \tag{4.13}$$

Next the total wave dissipation, $\bar{D}$, is distributed proportionally over the wave directions:

$$D(x,y,\theta) = \frac{S_w(x,y,\theta)}{E_w(x,y)} \bar{D} \tag{4.14}$$

This closes the set of equations for the wave action balance. Given the spatial distribution of the wave action and therefore wave energy the wave forcing can be calculated utilizing the radiation stress tensor:

$$
\begin{aligned}
F_x &= -\left(\frac{\partial S_{xx}}{\partial x} + \frac{\partial S_{xy}}{\partial y} \right) \\
F_y &= -\left(\frac{\partial S_{xy}}{\partial x} + \frac{\partial S_{yy}}{\partial y} \right)
\end{aligned}
\tag{4.15}
$$

And:

$$
\begin{aligned}
S_{xx} &= \int \left(\frac{c_g}{c} \left(1+\cos^2\theta\right) - \frac{1}{2} \right) S_w d\theta \\
S_{xy} &= S_{yx} = \int \sin\theta \cos\theta \left(\frac{c_g}{c} S_w \right) d\theta \\
S_{yy} &= \int \left(\frac{c_g}{c} \left(1+\sin^2\theta\right) - \frac{1}{2} \right) S_w d\theta
\end{aligned}
\tag{4.16}
$$

Similar to the solution of the shallow water equations we use an up-wind schematisation to solve the wave action balance. The wave action is given at the same points at the water level. The advection of wave action is then discretized as follows:

$$
\begin{aligned}
\frac{\partial c_x^n A^n}{\partial x}(\mathrm{i,j,k}) &= \frac{c_{x,i,j,k}^n A_{i,j,k}^n - c_{x,i-1,j,k}^n A_{i-1,j,k}^n}{x_{\mathrm{i,j}} - x_{\mathrm{i-1,j}}} \quad , c_{x,i,j,k}^n > 0 \\
\frac{\partial c_x^n A^n}{\partial x}(\mathrm{i,j,k}) &= \frac{c_{x,i+1,j,k}^n A_{i+1,j,k}^n - c_{x,i,j,k}^n A_{i,j,k}^n}{x_{\mathrm{i+1,j}} - x_{\mathrm{i,j}}} \quad , c_{x,i,j,k}^n < 0
\end{aligned}
\tag{4.17}
$$

$$
\begin{aligned}
\frac{\partial c_y^n A^n}{\partial y}(\mathrm{i,j,k}) &= \frac{c_{y,i,j,k}^n A_{i,j,k}^n - c_{y,i,j-1,k}^n A_{i,j-1,k}^n}{y_{\mathrm{i,j}} - y_{\mathrm{i,j-1}}} \quad , c_{y,i,j,k}^n > 0 \\
\frac{\partial c_y^n A^n}{\partial y}(\mathrm{i,j,k}) &= \frac{c_{y,i,j+1,k}^n A_{i,j+1,k}^n - c_{y,i,j,k}^n A_{i,j,k}^n}{y_{\mathrm{i,j+1}} - y_{\mathrm{i,j}}} \quad , c_{y,i,j,k}^n < 0
\end{aligned}
\tag{4.18}
$$

$$
\begin{aligned}
\frac{\partial c_\theta^n A^n}{\partial \theta}(\mathrm{i,j,k}) &= \frac{c_{\theta,i,j,k}^n A_{i,j,k}^n - c_{\theta,i,j,k-1}^n A_{i,j,k-1}^n}{\theta_{\mathrm{i,j,k}} - \theta_{\mathrm{i,j,k-1}}} \quad , c_{\theta,i,j,k}^n > 0 \\
\frac{\partial c_\theta^n A^n}{\partial \theta}(\mathrm{i,j,k}) &= \frac{c_{\theta,i,j,k+1}^n A_{i,j,k+1}^n - c_{\theta,i,j,k}^n A_{i,j,k}^n}{\theta_{\mathrm{i,j,k+1}} - \theta_{\mathrm{i,j,k}}} \quad , c_{\theta,i,j,k}^n < 0
\end{aligned}
\tag{4.19}
$$

Similar for the wave action balance:

$$
\frac{A_{i,j,k}^{n+1} - A_{i,j,k}^n}{\Delta t} = -\left.\frac{\partial c_x^n A^n}{\partial x}\right|_{i,j,k} - \left.\frac{\partial c_y^n A^n}{\partial y}\right|_{i,j,k} - \left.\frac{\partial c_\theta^n A^n}{\partial \theta}\right|_{i,j,k} - \left.\frac{D}{\sigma}\right|_{i,j,k}
\tag{4.20}
$$

which yields the wave energy at the new time level.

4.2 Shallow water equations solver

Shallow water equations, neglecting Coriolis and horizontal diffusion terms, and (grey terms), for the moment, wind shear stress:

$$\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}=\frac{\tau_{sx}}{\rho h}-\frac{\tau_{bx}}{\rho h}-g\frac{\partial \eta}{\partial x}+\frac{F_x}{\rho h} \tag{4.21}$$

$$\frac{\partial v}{\partial t}+u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}=+\frac{\tau_{sy}}{\rho h}-\frac{\tau_{by}}{\rho h}-g\frac{\partial \eta}{\partial y}+\frac{F_y}{\rho h} \tag{4.22}$$

$$\frac{\partial \eta}{\partial t}+\frac{\partial hu}{\partial x}+\frac{\partial hv}{\partial y}=0 \tag{4.23}$$

Here, h is the water depth, u, v are velocities in x and and y direction, τ_{bx},τ_{by} are the bed shear stresses, g is the acceleration of gravity, η is the water level and F_x, F_y are the wave-induced stresses.

We apply an upwind schematisation, since the horizontal scale of the problem is limited and such a scheme deals with shocks in a natural way.

We apply a staggered grid, where bed levels and water levels are defined in the centre of cells, and velocity components at the cell interfaces.

If *nx,ny* are the number of cells in both directions, the water level points are numbered from 1 to *nx+1* and from 1 to *ny+1*.

The water level gradients are computed at the cell interfaces and are given by:

$$\frac{\partial \eta}{\partial x}(\mathrm{i,j})=\frac{\eta_{\mathrm{i+1,j}}-\eta_{\mathrm{i,j}}}{x_{\mathrm{i+1,j}}-x_{\mathrm{i,j}}} \tag{4.24}$$

$$\frac{\partial \eta}{\partial y}(\mathrm{i,j})=\frac{\eta_{\mathrm{i,j+1}}-\eta_{\mathrm{i,j}}}{x_{\mathrm{i,j+1}}-x_{\mathrm{i,j}}} \tag{4.25}$$

For computing the shear stresses at the cell interfaces we need the velocity magnitudes at these interfaces. These are composed by combining the normal velocity component at the interface and the average of the 4 adjacent tangential components:

$$\begin{aligned} v_{u,i,j} &= \frac{1}{4}(v_{i,j-1}+v_{i,j}+v_{i+1,j-1}+v_{i+1,j}) \\ u_{v,i,j} &= \frac{1}{4}(u_{i-1,j}+u_{i,j}+u_{i-1,j+1}+u_{i,j+1}) \end{aligned} \tag{4.26}$$

The water depth in each cell is computed as:

$$h_{i,j}=\eta_{i,j}-z_{b,i,j} \tag{4.27}$$

For the depth at cell interfaces, following Stelling and Duinmeijer (2003) we distinguish between the depth used in the continuity equation and that used in the momentum equation. The depth at the interfaces *for the continuity equation* is taken as the upwind depth in case the velocity is greater than a minimum velocity, or the average depth between the adjacent cells in case the velocity is less than this minimum velocity:

$$\begin{aligned} h_{u,i,j} &= h_{i,j} && ,\ u_{i,j} > u_{\min} \\ h_{u,i,j} &= h_{i+1,j} && ,\ u_{i,j} < -u_{\min} \\ h_{u,i,j} &= \frac{1}{2}(h_{i,j} + h_{i+1,j}) && ,\ \left|u_{i,j}\right| < u_{\min} \end{aligned} \tag{4.28}$$

$$\begin{aligned} h_{v,i,j} &= h_{i,j} && ,\ v_{i,j} > v_{\min} \\ h_{v,i,j} &= h_{i,j+1} && ,\ v_{i,j} < -v_{\min} \\ h_{v,i,j} &= \frac{1}{2}(h_{i,j} + h_{i,j+1}), && \left|v_{i,j}\right| < v_{\min} \end{aligned} \tag{4.29}$$

For the depth *in the momentum balance* we take the average depth between the cell centers:

$$h_{mu,i,j} = \frac{1}{2}(h_{i,j} + h_{i+1,j}) \tag{4.30}$$

$$h_{mv,i,j} = \frac{1}{2}(h_{i,j} + h_{i,j+1}), \tag{4.31}$$

The advection terms in x-direction are approximated as follows:

$$\begin{aligned} u\frac{\partial u}{\partial x}^{n}_{i,j} &= \frac{1}{2}\frac{h_{u,i,j}u_{i,j} + h_{u,i-1,j}u_{i-1,j}}{h_{mu,i,j}}\ \frac{u^n_{i,j} - u^n_{i-1,j}}{x^n_{i,j} - x^n_{i-1,j}} && ,u^n_{i,j} > 0 \\ u\frac{\partial u}{\partial x}^{n}_{i,j} &= \frac{1}{2}\frac{h_{u,i,j}u_{i,j} + h_{u,i+1,j}u_{i+1,j}}{h_{mu,i,j}}\ \frac{u^n_{i+1,j} - u^n_{i,j}}{x^n_{i+1,j} - x^n_{i,j}} && ,u^n_{i,j} < 0 \end{aligned} \tag{4.32}$$

$$v\frac{\partial u}{\partial y}^{n}_{i,j} = v^{\,n}_{u,i,j}\frac{u^n_{i,j+1} - u^n_{i,j-1}}{y^n_{i,j+1} - y^n_{i,j-1}} \tag{4.33}$$

The advection terms in y-direction are approximated as follows:

$$\begin{aligned} v\frac{\partial v}{\partial y}^{n}_{i,j} &= \frac{1}{2}\frac{h^n_{v,i,j}v^n_{i,j} + h^n_{v,i,j-1}v^n_{i,j-1}}{h^n_{mv,i,j}}\ \frac{v^n_{i,j} - v^n_{i,j-1}}{y^n_{i,j} - y^n_{i,j-1}} && ,v^n_{i,j} > 0 \\ v\frac{\partial v}{\partial y}^{n}_{i,j} &= \frac{1}{2}\frac{h^n_{v,i,j}v^n_{i,j} + h^n_{v,i,j+1}v^n_{i,j+1}}{h^n_{mv,i,j}}\ \frac{v^n_{i,j+1} - v^n_{i,j}}{y^n_{i,j+1} - y^n_{i,j}} && ,v^n_{i,j} < 0 \end{aligned} \tag{4.34}$$

$$u\frac{\partial v}{\partial x}^{n}_{i,j} = u^{\,n}_{v,i,j}\frac{v^n_{i+1,j} - v^n_{i+1,j}}{x^n_{i+1,j} - x^n_{i+1,j}} \tag{4.35}$$

The momentum equation is discretized as follows:

$$\frac{u_{i,j}^{n+1}-u_{i,j}^{n}}{\Delta t}=-u\frac{\partial u^{n}}{\partial x}_{i,j}-v\frac{\partial u^{n}}{\partial y}_{i,j}-\frac{g\,u_{i,j}^{n}\sqrt{u_{i,j}^{n\ 2}+v_{u,i,j}^{n\ \ 2}}}{h_{u,i,j}^{n}C^{2}}-g\frac{\eta_{\mathrm{i+1,j}}^{n}-\eta_{\mathrm{i,j}}^{\mathrm{n}}}{x_{\mathrm{i+1,j}}-x_{\mathrm{i,j}}}+\frac{F_{x,i,j}}{\rho h_{u,i,j}} \quad (4.36)$$

$$\frac{v_{i,j}^{n+1}-v_{i,j}^{n}}{\Delta t}=-v\frac{\partial v^{n}}{\partial y}_{i,j}-u\frac{\partial v^{n}}{\partial x}_{i,j}-\frac{g\,v_{i,j}^{n}\sqrt{u_{v,i,j}^{n\ \ 2}+v_{i,j}^{n\ 2}}}{h_{v,i,j}^{n}C^{2}}-g\frac{\eta_{\mathrm{i,j+1}}-\eta_{\mathrm{i,j}}}{y_{\mathrm{i,j+1}}-y_{\mathrm{i,j}}}+\frac{F_{y,i,j}}{\rho h_{v,i,j}} \quad (4.37)$$

From this, the velocities at the new time step level are computed. The water level is then updated by:

$$\frac{\eta_{i,j}^{n+1}-\eta_{i,j}^{n}}{\Delta t}=-\frac{u_{i,j}^{n+1}h_{i,j}^{n}-u_{i-1,j}^{n+1}h_{i-1,j}^{n}}{x_{\mathrm{u,i,j}}-x_{\mathrm{u,i-1,j}}}-\frac{v_{i,j}^{n+1}h_{\mathrm{i,j}}-v_{i,j-1}^{n+1}h_{i,j-1}^{n}}{y_{\mathrm{v,i,j}}-y_{\mathrm{v,i,j-1}}} \quad (4.38)$$

Generalized Lagrangian Mean formulation

To account for the wave induced mass-flux and the subsequent (return) flow the shallow water equations are cast into a Generalized Lagrangian Mean (GLM) formulation (Walstra et al, 2000). To that end the Eulerian shallow water velocity u^E is replaced with its lagrangian equivalent, u^L:

$$u^{L}=u^{E}+u^{S}\quad and\quad v^{L}=v^{E}+v^{S} \quad (4.39)$$

and u^S , v^S represents the Stokes drift in x- and y-direction respectively (Phillips, 1977):

$$u^{S}=\frac{E_{w}\cos\theta}{\rho hc}\quad and\quad v^{S}=\frac{E_{w}\sin\theta}{\rho hc} \quad (4.40)$$

where the wave-group varying short wave energy and direction are obtained from the wave-action balance. The resulting GLM-momentum equations are given by:

$$\begin{aligned}\frac{\partial u^{L}}{\partial t}+u^{L}\frac{\partial u^{L}}{\partial x}+v^{L}\frac{\partial u^{L}}{\partial y}&=-\frac{\tau_{bx}^{E}}{\rho h}-g\frac{\partial\eta}{\partial x}+\frac{F_{x}}{\rho h}\\ \frac{\partial v^{L}}{\partial t}+u^{L}\frac{\partial v^{L}}{\partial x}+v^{L}\frac{\partial v^{L}}{\partial y}&=-\frac{\tau_{by}^{E}}{\rho h}-g\frac{\partial\eta}{\partial y}+\frac{F_{y}}{\rho h}\end{aligned} \quad (4.41)$$

for the x- and y-direction respectively. This operation shows that the GLM equations for the depth-averaged flow are very similar to the previously described Eulerian formulation, with the exception of the bottom shear stress terms that are calculated with the Eulerian velocities as experienced by the bed:

$$u^{E}=u^{L}-u^{S}\quad and\quad v^{E}=v^{L}-v^{S} \quad (4.42)$$

and not with the GLM velocities. Also, the boundary condition for the flow computations has to be expressed in functions of (u^L , v^L) and not (u^E, v^E).

4.3 Sediment transport

Advection-diffusion scheme

The sediment transport is modeled with a depth-averaged advection diffusion equation [Gallapatti, 1983]:

$$\frac{\partial hC}{\partial t}+\frac{\partial hCu^E}{\partial x}+\frac{\partial hCv^E}{\partial y}+\frac{\partial}{\partial x}\left[D_h h\frac{\partial C}{\partial x}\right]+\frac{\partial}{\partial y}\left[D_h h\frac{\partial C}{\partial y}\right]=\frac{hC_{eq}-hC}{T_s} \quad (4.43)$$

where C represents the depth-averaged sediment concentration which varies on the infragravity time scale. The entrainment of the sediment is represented by an adaptation time T_s, given by a simple approximation based on the local water depth, h, and sediment fall velocity w_s:

$$T_s = \max\left(0.05\frac{h}{w_s},0.2\right)s \quad (4.44)$$

where a small value of T corresponds to nearly instantaneous sediment response. The entrainment or deposition of sediment is determined by the mismatch between the actual sediment concentration and the equilibrium concentration, C_{eq}, thus representing the source term in the sediment transport equation.

The differential equations for the advection diffusion of sediment is solved with finite differences using the first order up-wind scheme discussed earlier with the water depths at the old time level and the corresponding velocities at the new time level. The horizontal x-advection is then given by:

$$\begin{aligned}
\left(\frac{\partial hCu^E}{\partial x}\right)_{i,j} &= \frac{\left(h^nC^nu^{E,n+1}\right)_{i,j}-\left(h^nC^nu^{E,n+1}\right)_{i-1,j}}{x_{i,j}-x_{i-1,j}} \qquad ,u^{E,n+1}_{i,j}>0\\
\left(\frac{\partial hCu^E}{\partial x}\right)_{i,j} &= \frac{\left(h^nC^nu^{E,n+1}\right)_{i+1,j}-\left(h^nC^nu^{E,n+1}\right)_{i,j}}{x_{i+1,j}-x_{i,j}} \qquad ,u^{E,n+1}_{i,j}<0
\end{aligned} \quad (4.45)$$

a similar expression for the horizontal advection in the y-direction:

$$\left(\frac{\partial hCv^E}{\partial y}\right)_{i,j} = \frac{\left(h^n C^n v^{E,n+1}\right)_{i,j} - \left(h^n C^n v^{E,n+1}\right)_{i-1,j}}{y_{i,j} - y_{i,j-1}} \qquad , v_{i,j}^{E,n+1} > 0$$

$$\left(\frac{\partial hCv^E}{\partial y}\right)_{i,j} = \frac{\left(h^n C^n v^{E,n+1}\right)_{i,j+1} - \left(h^n C^n v^{E,n+1}\right)_{i,j}}{y_{i,j+1} - y_{i,j}} \qquad , v_{i,j}^{E,n+1} < 0 \tag{4.46}$$

The horizontal diffusion is evaluated at the old time level *n* and approximated by:

$$\left(\frac{\partial}{\partial x}\left(D_H h \frac{\partial C}{\partial x}\right)\right)_{i,j} = \frac{\left(D_H h C_{\partial x}\right)_{i+1,j} - \left(D_H h C_{\partial x}\right)_{i,j}}{x_{i+1,j} - x_{i,j}} \tag{4.47}$$

Where the cross-shore gradient in the sediment concentration is given by:

$$C_{\partial x} = \left(\frac{\partial C}{\partial x}\right)_{i,j} = \frac{C_{i+1,j} - C_{i,j}}{x_{i+1,j} - x_{i,j}} \tag{4.48}$$

And similarly for the y-direction:

$$\left(\frac{\partial}{\partial y}\left(D_H h \frac{\partial C}{\partial y}\right)\right)_{i,j} = \frac{\left(D_H h C_{\partial y}\right)_{i,j+1} - \left(D_H h C_{\partial y}\right)_{i,j}}{y_{i,j+1} - y_{i,j}} \qquad , v_{i,j}^{E} < 0 \tag{4.49}$$

Where the along-shore gradient in the sediment concentration, C_y , is given by:

$$\left(\frac{\partial C}{\partial y}\right)_{i,j} = \frac{C_{i,j+1} - C_{i,j}}{y_{i,j+1} - y_{i,j}} \tag{4.50}$$

The time up-date of the sediment concentration is then given by:

$$\begin{aligned} &\frac{h_{i,j}^{n+1} C_{i,j}^{n+1} - h_{i,j}^{n} C_{i,j}^{n}}{\Delta t} + \left[\frac{\partial hCu^E}{\partial x}\right]_{i,j}^{n} + \left[\frac{\partial hCv^E}{\partial y}\right]_{i,j}^{n} + \\ &+ \left[\frac{\partial}{\partial x}\left[D_h h \frac{\partial C}{\partial x}\right]\right]_{i,j}^{n} + \left[\frac{\partial}{\partial y}\left[D_h h \frac{\partial C}{\partial y}\right]\right]_{i,j}^{n} = \left[\frac{hC_{eq} - hC}{T_s}\right]_{i,j}^{n} \end{aligned} \tag{4.51}$$

The bed-update is discussed next. Based on the gradients in the sediment transport the bed level changes according to:

$$\frac{\partial z_b}{\partial t} + \frac{\partial S_x}{\partial x} + \frac{\partial S_y}{\partial y} = 0 \tag{4.52}$$

where S_x and S_y represent the sediment transport rates in *x*- and *y*-direction respectively, given by:

$$S^n_{x,i,j} = \left[\frac{\partial hCu^E}{\partial x}\right]^n_{i,j} + \left[\frac{\partial}{\partial x}\left[D_h h \frac{\partial C}{\partial x}\right]\right]^n_{i,j} \tag{4.53}$$

and

$$S^n_{y,i,j} = \left[\frac{\partial hCv^E}{\partial y}\right]^n_{i,j} + \left[\frac{\partial}{\partial y}\left[D_h h \frac{\partial C}{\partial y}\right]\right]^n_{i,j} \tag{4.54}$$

The bed-update is then approximated by:

$$\frac{z^{n+1}_{b,i,j} - z^n_{b,i,j}}{\Delta t} + f_{mor}\left[\frac{S^n_{x,i,j} - S^n_{x,i-1,j}}{\Delta x} + \frac{S^n_{y,i,j} - S^n_{y,i,j-1}}{\Delta y}\right] = 0 \tag{4.55}$$

where f_{mor} represents a morphological factor to speed up the bed evolution.

Transport formulations

The equilibrium sediment concentration can be calculated with various sediment transport formulae. At the moment the sediment transport formulation of Soulsby-van Rijn (Soulsby, 1997) has been implemented. The C_{eq} is then given by :

$$C_{eq} = \frac{A_{sb} + A_{ss}}{h}\left(\left(|u^E|^2 + 0.018\frac{u^2_{rms}}{C_d}\right)^{0.5} - u_{cr}\right)^{2.4}(1 - \alpha_b m) \tag{4.56}$$

where sediment is stirred by the Eulerian mean and infragravity velocity in combination with the near bed short wave orbital velocity obtained from the wave-group varying wave energy. The combined mean/infrgarvity and orbital velocity have to exceed a threshold value, u_{cr}, before sediment is set in motion. The drag coefficient, C_d, is due to flow velocity only (ignoring short wave effects). To account for bed-slope effects on the equilibrium sediment concentration a bed-slope correction factor is introduced, where the bed-slope is denoted by m and α_b represents a calibration factor. The bed load coefficients A_{sb} and the suspended load coefficient A_{ss} are functions of the sediment grain size, relative density of the sediment and the local water depth (see Soulsby [1997] for details).

4.4 Bottom updating

Avalanching

To account for the slumping of sandy material during storm-induced dune erosion avalanching is introduced to update the bed-evolution. Avalanching is introduced when a critical bed-slope is exceeded:

$$\left|\frac{\partial z_b}{\partial x}\right| > m_{cr} \tag{4.57}$$

Where the estimated bed slope is given by:

$$\frac{\partial z_b}{\partial x} = \frac{z_{b,i+1,j} - z_{b,i,j}}{\Delta x} \tag{4.58}$$

The bed-change within one time step is then given by:

$$\begin{aligned}
\Delta z_b &= \min\left(\left(\left|\frac{\partial z_b}{\partial x}\right| - m_{cr} \right)\Delta x, \ \ 0.005 \right) \quad ,\frac{\partial z_b}{\partial x} > 0 \\
\Delta z_b &= \max\left(-\left(\left|\frac{\partial z_b}{\partial x}\right| - m_{cr} \right)\Delta x, -0.005 \right) \quad ,\frac{\partial z_b}{\partial x} < 0
\end{aligned} \tag{4.59}$$

where a threshold of 0.005 m has been introduced to prevent the generation of large shockwaves. The corresponding bed update is given by:

$$\begin{aligned}
z_{b,i,j}^{n+1} &= z_{b,i,j}^{n} + \Delta z_{b,i,j} \\
z_{b,i+1,j}^{n+1} &= z_{b,i+1,j}^{n} - \Delta z_{b,i,j}
\end{aligned} \tag{4.60}$$

To account for continuity, e.g. when sand is deposited within the wet part of the domain, the water level is also updated:

$$\begin{aligned}
z_{s,i,j}^{n+1} &= z_{s,i,j}^{n} + \Delta z_{b,i,j} \\
z_{s,i+1,j}^{n+1} &= z_{s,i+1,j}^{n} - \Delta z_{b,i,j}
\end{aligned} \tag{4.61}$$

Similar expressions are used for the subsequent avalanching in the y-direction.

5 XBeach validation

5.1 Long wave propagation and numerical damping

The purpose of the first test is to check if the scheme is not too dissipative and that it does not create large errors in propagation speed.

A long wave with a small amplitude of 0.01 m and period of 80 s was sent into a domain of 5 m depth, grid size of 5 m and a length of 1 km. At the end, a fully reflecting wall is imposed. The wave length in this case should be sqrt(9.81*5)*80 = 560 m. The velocity amplitude should be sqrt(g/h)*amp = sqrt(9.81/5)*0.01 = 0.014 m. After the wave has reached the wall, a standing wave with double amplitude should be created.

As Figures 5.1 and 5.2 show, the model accurately represents this situation. There is hardly any dissipation, the wave length is very close to what it should be and there is no re-reflection off the seaward boundary.

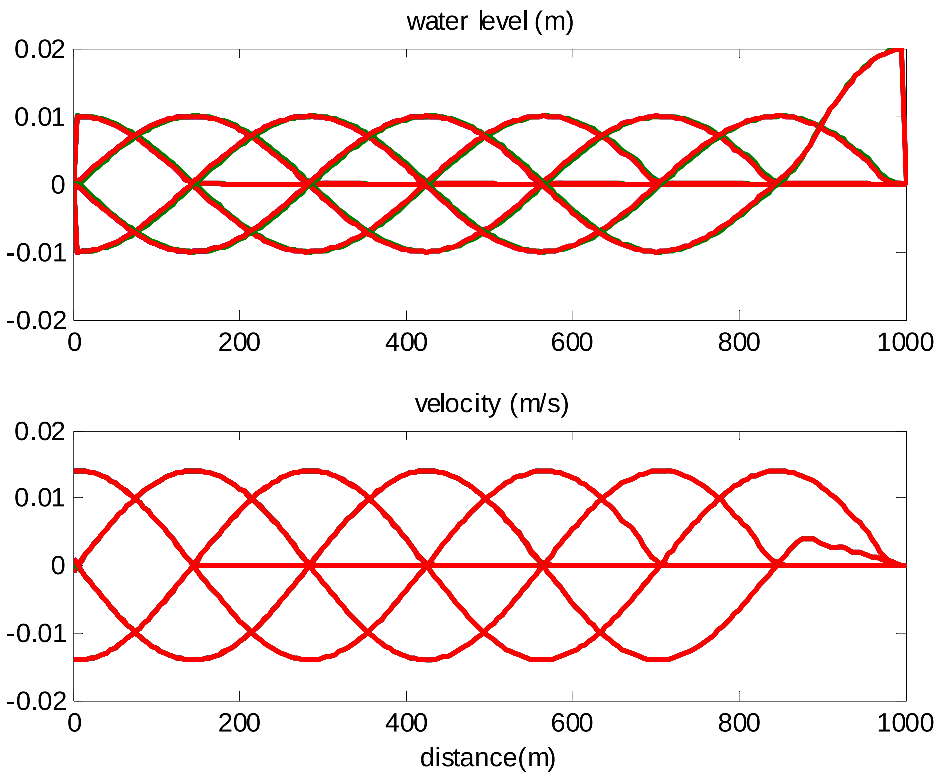

Figure 5.1 Snapshots of water level and velocity at T/4 intervals; just as wave hits wall.

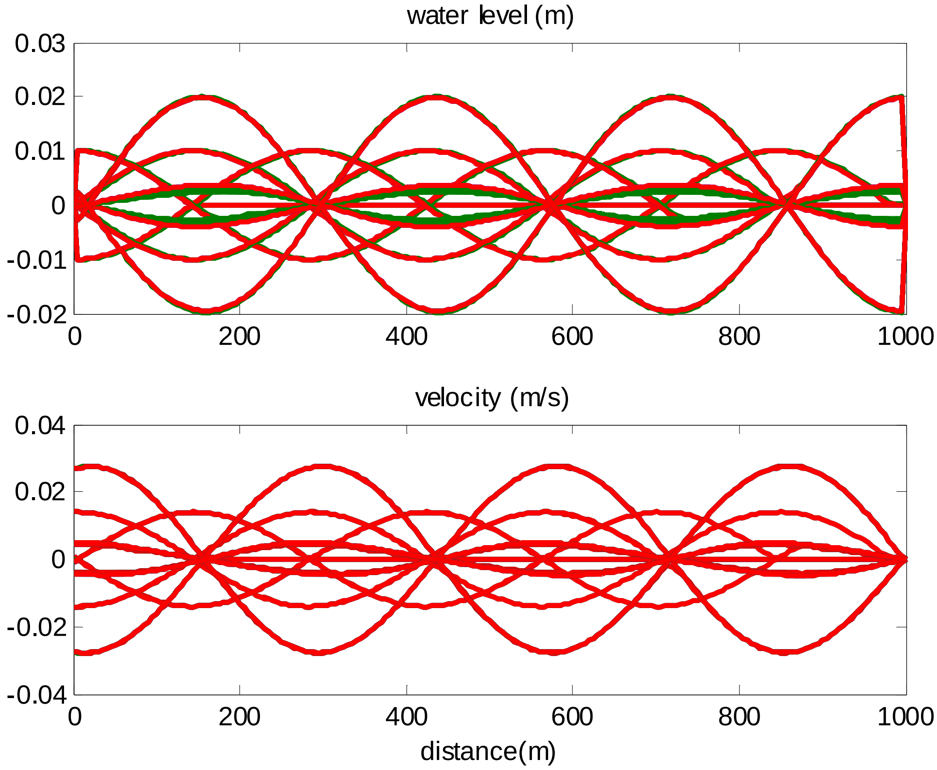

Figure 5.2 Same as 5.1, after long time.

5.2 Long wave runup on sloping beach; comparison with Carrier and Greenspan (1958)

The purpose of this test is to check the ability of the model to represent runup and rundown of long waves. To this end, a comparison was made with the analytical solution by Carrier and Greenspan (1958), which describes the motion of harmonic, non-breaking long waves on a plane sloping beach without friction.

In Figure 5.3 the model results for waves at an amplitude of ½ the breaking wave amplitude are shown, at 1/20 T intervals. The agreement is quite good, though there is very small disturbance/lag during rundown. Typical of the solution is that the profiles during runup and rundown should be exactly equal; apart from a small area near the water line during rundown, this is the case.

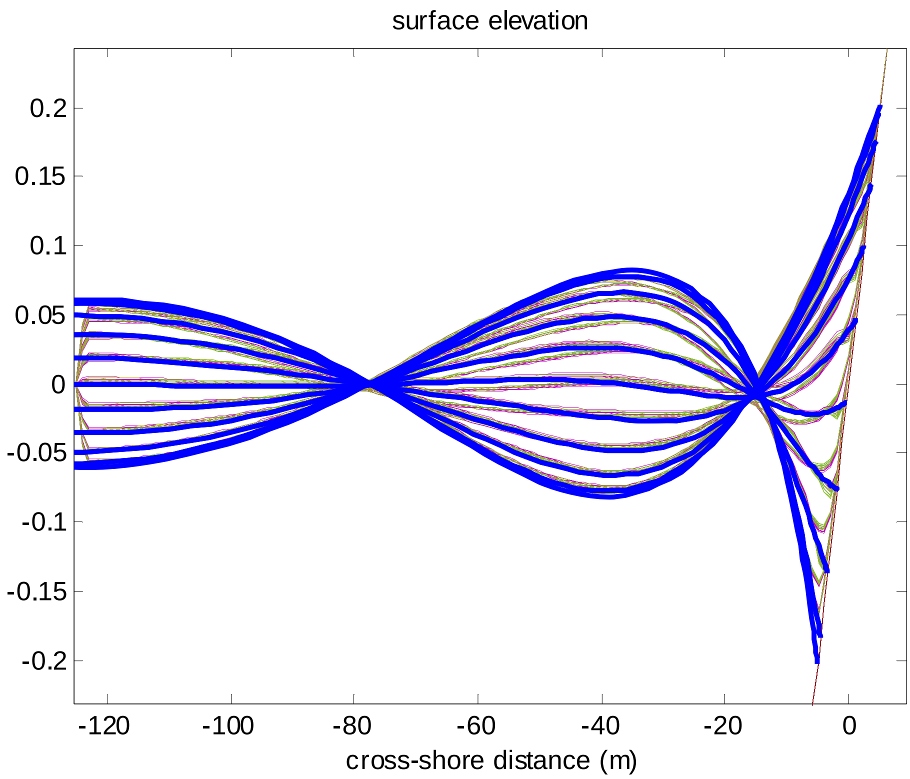

Figure 5.3 Surface elevation snapshots at 1/20 T intervals; model (thin brown lines) and analytical solution (thick blue lines).

5.3 Stationary wave propagation. dissipation and setup

The purpose of this test was to check the wave energy and momentum balance in stationary mode.

The Delta Flume test of Arcilla et al. (1993), test 2E was used. This test was carried out with an increased water level and significant dune erosion occurred during the test. Besides, extensive hydrodynamic, sediment transport and morphological measurements were carried out.

The dissipation model of Baldock (1998) was applied, with a gamma value of 0.8. The results show a good agreement for the Hrms wave height, mean setup and rms value of the orbital velocity.

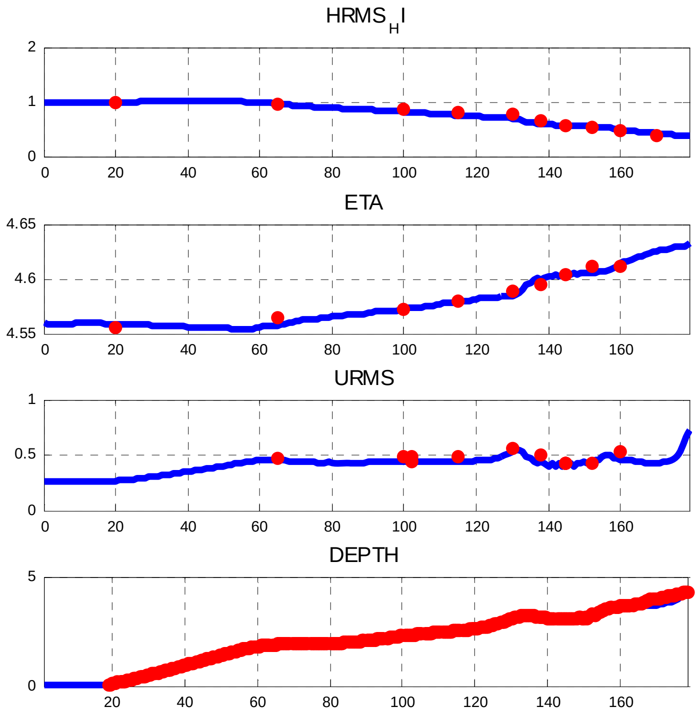

Figure 5.4 Wave height decay, setup and orbital velocity, modelled in stationary mode (drawn blue line) vs. measured (red dots). LIP 11D test 2B.

5.4 Instationary surf zone flows in large-scale flume test

The purpose of this test was to verify the hydrodynamics of the model when run in instationary mode, viz. with a time-varying wave energy imposed at the offshore boundary.

The same test case as before was used. The time series of wave energy was generated by a procedure described in Roelvink (1993b), based on a JONSWAP spectral shape as was applied in the test. Zero-order steering was applied in the flume test; therefore no incident bound long wave was imposed in the numerical experiment.

The results are given in Figures 5.5 and 5.6, and show that the short wave decay and the generated long waves are represented quite well, both in surface elevation and in near-bottom velocity. Also the time-averaged water level is represented quite accurately.

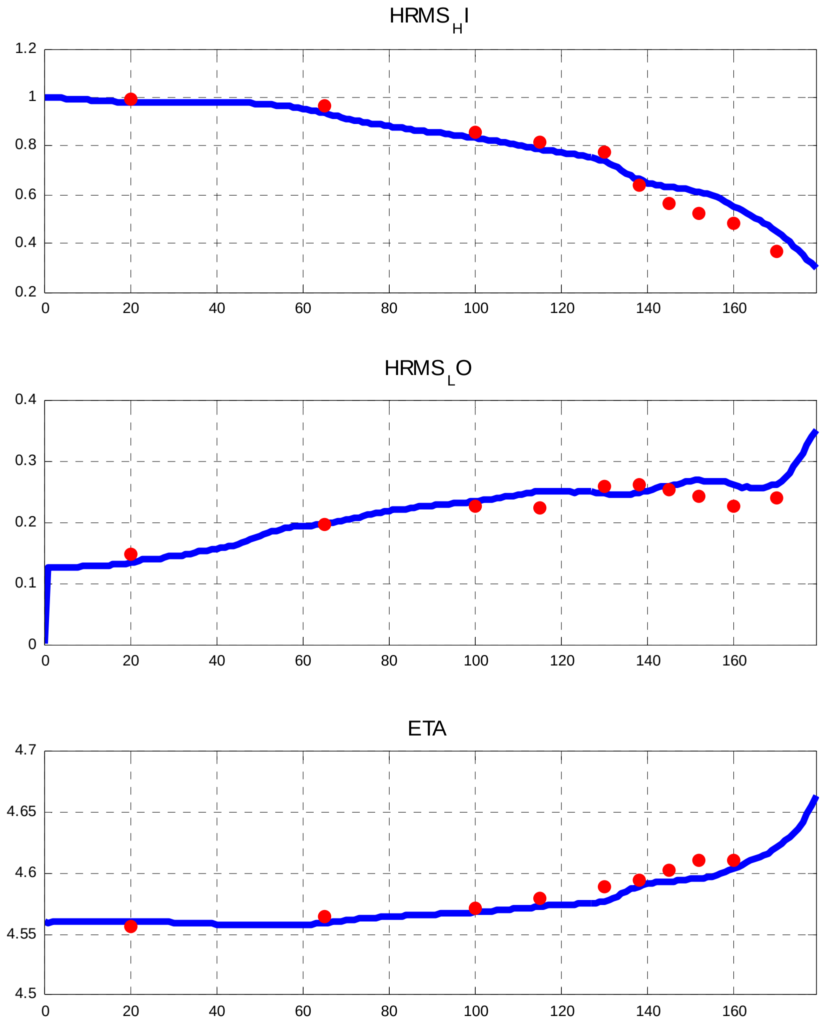

Figure 5.5 Wave height, LF wave height and setup, instationary model results vs measurements from LIP 11D, test 2E.

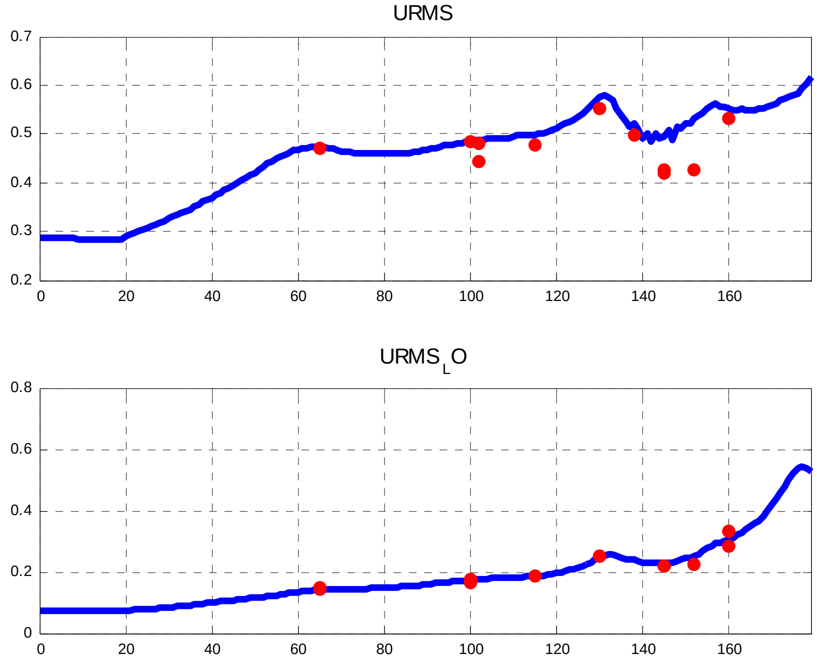

Figure 5.6 Orbital velociity and long wave rms velocity , instationary model results vs LIP 11D test 2E.

5.5 Dune erosion in large-scale flume test.

The purpose of this test was to verify the dune erosion modelling in instationary mode.

The model was run for 0.8 hours of hydrodynamic time with a morphological factor of 10, effectively representing a morphological simulation time of 8 hours.
A key element in the modelling is the avalanching algorithm; although the surfbeat waves that are explicitly modelled run up and down the upper beach, without a mechanism to transport sand from dry to wet the dune erosion process will not happen. A relatively simple approach, whereby an underwater critical slope of 0.3 and a critical slope above water of 1.0 were applied, proves to be quite successful in representing the retreat of the upper beach and dune face. A grid resolution of 1 m was applied. In Figure 5.7 The measured and modelled bed evolution is shown, which looks quite promising in the upper region. The behaviour of the bar at approx. 135 m is not represented well; for this, additional processes such as the effect of surface rollers and wave asymmetry/skewness have to be taken into account.

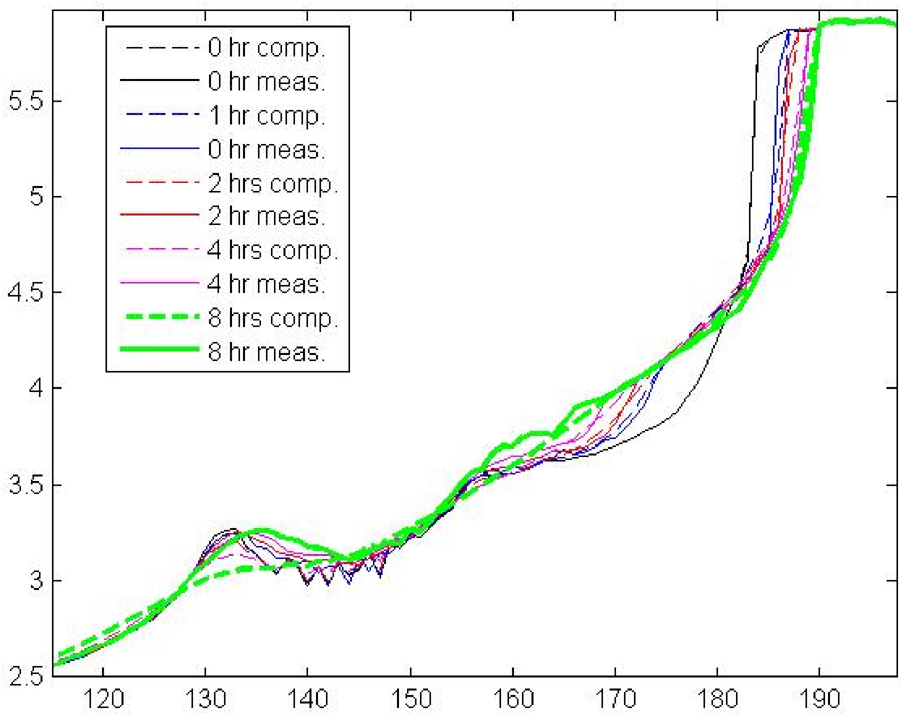

Figure 5.7 Measured and modelled bed level after 0, 1, 2, 4 and 8 hours of wave action, for a water level of 5. 1 m above the flume bottom.

6 Conclusions and next steps

Significant progress has been made over the last three months in modelling the dune erosion process, and a first series of validation tests has been carried out. The approach tested so far has been to run in instationary mode, where wave-group generated long waves are generated which represent the dominant motion in the swash zone. As these motions are also likely to dominate overwashing processes we are eager to proceed to test cases where overwashing occurs.

A drawback of this approach may be the required resolution, in the order of metres; in 1D simulations this is no problem, in 2DH simulations covering larger domains it may become restrictive, though parallellization can solve much of this problem. Still, it is worth considering alternative ‘quick-and-dirty’ approaches based on stationary wave and flow modelling combined with an extrapolation method for the actual dune erosion.

Our plans for the coming months are summarized below:

- Investigate Steetzel’s formulations, Van Rijn’s latest method
- Investigate original D3D “extrapolation method”
- Include roller model and wave asymmetry effects
- 2D offshore absorbing boundary conditions (Ap and Jamie)
- Include Q3D description of flow cf Reniers et al, 2004b
- Create a collaborative workspace on the IHE server.
- Start version management through the same server.
- Incorporate all cases in Delft Hydraulics test bed format
- Review formulations of wave impact contributions by short waves and long waves (e.g. Overton and Fisher).
- Future tests for testbed:
    - New dune erosion test from Deltaflume (available)
    - Berm test (available)
    - Scheveningen berm test in Deltaflume (this Fall)
    - Oregon test (after summer)
    - 2D tests are only in situ but with little data. Possible: 2D Dauphin, Monterey, East coast Florida. Brad will check with Abby Sallenger, Ap will contact Brad. Ad to check with Thornton, DJ to check with CPI. NCEX data is also possibility, but little coastal damage. Nobu Kobayashi at Deleware University is proposing to NSF-International to conduct a lab study at PARI Japan or at ORST Tsunami tank. Ap will keep in contact.
- Dano Roelvink will go to MORPHOS meeting in Vicksburg at November 1-2.

7 References

Arcilla, A.S., Roelvink, J.A., O'Connor, B.A. Reniers, A., and Jimenez. J.A. The Delta Flume '93 Experiment. Coastal Dynamics '94. Arcilla, Stive and Kraus (eds), ASCE, New York, pp. 488-502.

Carrier, G.F. and H.P. Greenspan (1958). Water waves of finite amplitude on a sloping beach. J. Fluid Mech., Vol. 4, pp. 97-109.

Galapatti, R., 1983. A depth integrated model for suspended transport. Report 83-7, Communications on Hydraulics, Department of Civil Engineering, Delft University of Technology.

Reniers, A.J.H.M., J.A. Roelvink and E.B. Thornton. (2004). Morphodynamic modelling of an embayed beach under wave group forcing. J. of Geophysical Res. , VOL. 109, C01030, doi:10.1029/2002JC001586, 2004

Reniers, A.J.H.M., E.B. Thornton, T. Stanton and J.A. Roelvink. (2004b) Vertical flow structure during Sandy Duck: Observations and Modeling. Coastal Engineering, Volume 51, Issue 3, May 2004, Pages 237-260

Roelvink, J.A. (1993a) Dissipation in random wave groups incident on a beach. Coastal Eng., 19 (1993) pp. 127-150.

Roelvink, J.A. (1993b) Surf beat and its effect on cross-shore profiles. Ph.D. thesis, Delft University of Technology.

Soulsby, R. (1997). Dynamics of marine sands. Thomas Telford Publications, London, ISBN 0 7277 2584 X.

Stelling GS, Duinmeijer SPA. A staggered conservative scheme for every Froude number in rapidly varied shallow water flows. International Journal for Numerical Methods in Fluids. 2003; 43:1329-1354.